IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 3, March 2018

Novel Dynamic Structure to Implement Public Auditing Protocol on Cloud Data

Ch Sai Pavan[1], P Sudheer Kumar[2]

UG Student, Department of Computer Science & Engineering, VVIT, Guntur[1]

Asst Prof, Department of Computer Science & Engineering, VVIT, Guntur[2]

Abstract: As cloud computing is one of the rapidly growing technologies, cloud storage has been accepted by many people and organizations because it is a more convenient and outsourcing application which is high on-demand. Data owner generally thinks that whether his data is secure or not.So, many research people started designing auditing protocols which concentrates on outsourced data. In this paper, we propose a public auditing protocol with global and sampling block-less verification, batch auditing where data dynamics is more efficiently supported. It is important to note that novel dynamic structure in our protocol consists Doubly Linked Info Table (DLIT) and Location Array (LA).By using this dynamic structure, computational and communicational overheads are reduced, security is achieved. Moreover, numerical analysis and real-world experimental results illustrates that the proposed auditing protocol is going to be more efficient.

Keywords: Double Linked Info Table, Location Array, Auditing Protocol.

I. INTRODUCTION

Cloud storage, a service offered by the cloud has made data outsourcing to the cloud as an emerging trend. The development of cloud service is due to its high on-demand outsourcing function, ubiquitous network access and location-independent resources [1].However, data outsourced to the cloud are not kept securely. It will suffer from variety of security attacks both external and internal [2] and most commonly malicious network attacks which are more familiar to internet users that threatens the cloud data. It opens a door for the hackers, they might steal the data or manipulate the data which results in destroying the cloud's confidentiality, integrity and also availability. The outsourced data may also suffer from the cloud service provider's(CSP) illegal behavior i.e.., a CSP could secretly delete some storage area from the specified storage area in order to save space for other client's data following a CSP might attempt to obtain the outsourced data. Thus the confidentiality and integrity of the outsourced data is indanger. Hence, it results in designing an efficient auditing protocol that was concerns. Cloud storage in cloud computing. By using this protocol, a DO could remotely verifies that whether the data in cloud is stored correctly or not. To verify, a new entity called third-party auditor(TPA) was introduced, which executes the data, after accepting the auditing delegation from the respected DO.It is widely accepted, as it decreases the burden on the client side. Once the data are outsourced, it will remain unchanged [5], unless a DO might demand to update data with insert, update, modify, delete and other commands. Though, public auditing protocols with dynamic support which concentrates on cloud data results in developing an efficient protocol. There exists some protocols, which are too expensive and may offer low efficiency in supporting data dynamics. In Tian's paper [6], Dynamic Hash Table (DHT) is designed to support data dynamics which uses single linked sequence table. Though his efficient auditing scheme based on two-dimensional data structure has some drawbacks like CSP is going to suffer from collusion attacks from both the DO and TPA and there is no word of index switcher which maintains relationship between index and sequence no's of a data block following computational cost of the protocol is relatively high. So, inspired by [6], here we introduced a novel dynamic structure which is composed of double linked info table and location array, which makes more effective.

II. RELATED WORK

Cloud Storage is one of the facility provided by the Cloud Computing. Branches like data auditing, privacy preservation and dynamic updating are the hot topics of the cloud computing.For auditing the data, we have many protocols which are divided into public and private protocols. In private auditing protocols, DO and the CSP are only the participating entities. In this, DO only possess the private key and also the task of auditing is done by the DO itself. It results in increasing the burden on the DO's side, who is not always facilitated with computing resources. Moreover, the CSP distrust the DO, because the auditing task is done by the DO itself.To solve these problems, a third-party entity called TPA, a trustworthy entity is introduced. Atniese et al [7], in 2007, introduced public auditing protocol which is widely accepted by the researchers. Later, a large number of protocols were designed based on the schema "challenge-proof-verify". In 2013, Wang et al, came up with a point that the public auditing protocol in [7] might leaks the data.As a result, the auditing protocol in "Privacy-Preserving public auditing for secure cloud storage", was designed to be privacy preserving scheme through the combination of homomorphic linear authenticator (HLA) andrandom masking

DOI 10.17148/IJARCCE.2018.7313

technique. Later, in 2015, Worku et al's "Secure and efficient privacy-preserving public auditing for cloud storage," proposed that privacy preserving auditing protocol by Wang et al, will no more preserves the signer's privacy. So, the privacy preserving auditing scheme, proposed by Worku et al was found to be better than the work of Wang. Further Yu et al [8] extended to key exposure-resistant protocols. In 2016, they gave the solution for the exposure of client's secret keys in "Enabling cloud storage auditing with key-exposure resistance". Subsequently, they improved and perfected the auditing protocol with key-exposure resistance in [8], which is performing much better.In 2008, Ateniese et al first proposed a partially Provable Data Possession (PDP) protocol, which provides PDP techniques. Inspired by Ateniese,Erwat et al's "Dynamic provable data possession", extended the protocol to support full dynamic storage by employing skip list, which brought improvement in efficient and privacy. Later, Wang et al, also focused on extending to support full dynamic but, by using Merkle Hash Tree, which was accepted by many researchers. However, both the schemes involves high communication costs for updating and verification process.In 2013, in order to reduce the computational and communicational cost overheads, Zhu et al [9], constructed the audit-service based on index-hash table. But, there is a problem which are update operations like insertion and deletion on an index-table, which faces difficulties in locating an element on an average.In 2016, to solve this problem, Tian et al [6], designed a public auditing protocol based on DHT and Jain et al [10], designed a protocol with an index-switcher. In [6], data information is stored in DHT by block numbers.However, there is a chance that CSP and the DO & TPA involves in collusion attacksbecause timestamps are generated by the DO and that too TPA serves to the DO only.In [10], an index-switcher, which indicates the relationship between block indices and tag indices will avoid the re-computation tasks that occurs in block update operations. This switcher consists of two tables instead of maintaining a huge structure but, it fails to maintain a connection between those two tables.

So, in contrast to the above solution, in this paper, we are providing an efficient public auditing protocol with novel dynamic structure consisting of doubly linked info table and a location array. The proposed protocol can also achieve the mutual trust between CSP and the DO because, global auditing is used to resist the collusion attacks from the DO's and the TPA's. The doubly linked info table will locate an element more quickly and location array will maintain good relation between the blocks and also keeps a track of the sequence of the blocks and their specific locations.

III. PROPOSED METHODOLOGY

In this paper, we are going to introduce an efficient public auditing protocol supported with novel dynamic structure for the data outsourced into the cloud which performs better than the existing techniques. By using efficient auditing protocol, we are going to achieve the following advantages:

1. Global and Sampling Verification.
2. Efficient data dynamics with a novel dynamic structure.
3. Public auditing, Block-less verification and Batch auditing.

This model contains three major entities namely the CSP, the DO and the TPA.CSP is the one who provides multiple cloud services to the cloud clients and CSP is also allows data owners to store their data in the cloud.DO may be either an individual or an organization will keep trust in the cloud for outsourcing their data. Communication with TPA can be done by any electronic devices like mobiles, computers, tablets etc.TPA is a trustworthy third party between the CSP and the DO. TPA has the knowledge of data verification so it can provide better auditingresults.

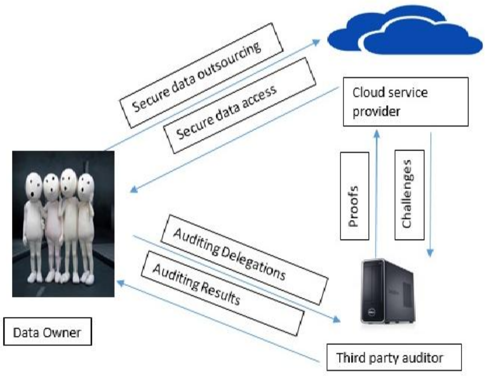


Fig 1: Architecture Diagram

The relation between these entities is like the DO outsources its data to the cloud and obtains them when needed. The connection between the DO and the CSP must be secure

DOI 10.17148/IJARCCE.2018.7313

and also encrypted. The TPA is chosen by the DO, who performs auditing on behalf of the DO. Whenever wants to verify certain data DO, DO would make an auditing delegation to the TPA. Upon receiving the request, TPA will send the auditing results to the CSP, in order to check the proofs and then to the DO.

A.The Proposed System

The dynamic structure is composed of a doubly linked info table and also a location array.

Doubly Linked Info Table:-It is a two-dimensional structure used by the TPA to store data, differing from the one-dimensional Index Hash Table. The structure of DLIT is shown in fig 2:The term (V $_{vn}$, $T_{u,\ f,\ n}$) in the fig: 2 indicates that for nth block of the uth user's file fth file, the timestamp for the V_{vn} th file version is T $_{u,f,n}$.The term ID_i in DLIT indicates file ID to identify each file uniquely.

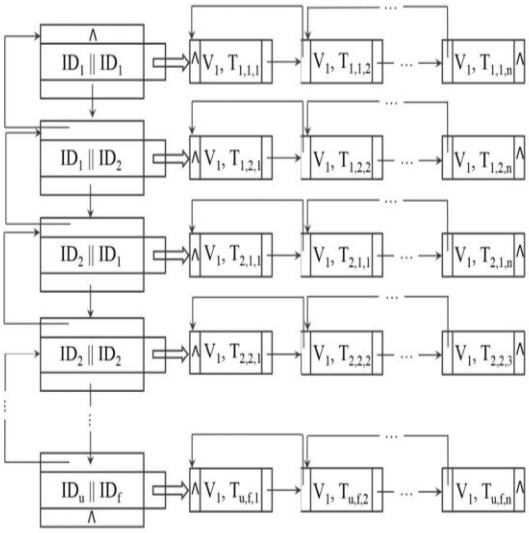


Fig 2: Doubly linked info table

Data in DLIT is divided into two parts namely file information and block information. The file information includes file ID along with user ID. In [6], [9], file information consists file ID only. When the number of files increases, it becomes more difficult to make file ID's unique. So, we use the concatenation of the file ID with user ID to identify each file, making unique identifier very easy. The right part is the block information, including the current version number and the timestamp, which are generated whenever a data block is uploaded or updated. In DLIT, both the file information and the block information are doubly linked, so they make a connection between forward, backward and previous, prior records also. By using doubly linked structure, insertion and deletion of a file or a block will never cause a change in other records. In addition to that, searching of an element is done at low cost.

Location Array:-It is used as an index switcher and also keeps the mapping between index numbers and serial numbers. For the specific operations like insertion and deletion of a block will affect its content, whereas search and modification will not.

$$\begin{pmatrix} A_{1,1} & A_{1,2} & A_{1,3} & A_{1,4} & A_{1,5} & \cdots\cdots \\ A_{2,1} & A_{2,2} & A_{2,3} & A_{2,4} & A_{2,5} & \cdots\cdots \\ A_{3,1} & A_{3,2} & A_{3,3} & A_{3,4} & A_{3,5} & \cdots\cdots \\ A_{4,1} & A_{4,2} & A_{4,3} & A_{4,4} & A_{4,5} & \cdots\cdots \\ A_{5,1} & A_{5,2} & A_{5,3} & A_{5,4} & A_{5,5} & \cdots\cdots \\ A_{6,1} & A_{6,2} & A_{6,3} & A_{6,4} & A_{6,5} & \cdots\cdots \end{pmatrix}$$

Fig 2.1: Location Array

B.The Proposed Protocol

In this paper, we are going to represent an efficient public auditing protocol with novel dynamic structure that consists of doubly linked info table and also location array.*Description of the protocol*:-The proposed auditing protocol is divided into two phases namely setup phase and verify phase. The setup phase is responsible for some preparation works and consists 3 algorithms: KeyGen, Filepro2C & FilePro2T. Similarly,the verifyphase also consists 3 algorithms: ChalGen, ProofGen, and VerifyProof.

1) *Setup phase*:-In this phase some pre-arrangements are made, which are the duties of DO. Then,some pre-processing tasks for the CSP and the DO are done by using FilePro2C and FilePro2T.

a) KeyGen: -The DO executes KeyGen to generate public & secret keys.First, the DO chooses the random signing key pair (ssk, spk) for the signature of the file name. Then, the DO picks onea (say), among set of non-negative integers that is less than the prime order of multi cyclic groups.Consequently, the secret key would be sk = (a, ssk), which is known by the DO. After that, DO picks random generators. Then, public key would be pk= {generators of cyclic groups,exponential value of secret key generated by the DO} In Short, the resultant of algorithm KeyGen would be {(secret key (sk), public key (pk)} = {(a, ssk), (u, g, v, spk)}. Where, u,g,v,spk refers to the generators of cyclic groups, exponential value of secret key generated by the DO

b) FilePro2C:- The DO executes the Filepro2C for pre-processing the files that are outsourced to the cloud. The DO divides the file into blocks. For instance, say 5 blocks

F= {m1, m2,….., m5}

Then, the DO generates the authenticator for each block by using hash function which requires version no and timestamp no of the blocks. Moreover, the DO generates the file tags based on ssk selected in KeyGen to ensure the integrity of the file info. Finally, the DO uploads the Unique File Information, authenticator and Blocks in File.At this point, pre-processing tasks for the CSP have all been completed.

c) FilePro2T:- The DO executes the FilePro2T to process the information handled by the TPA. The file information and the block information are stored in the DLIT where the specific block location is stored in the LA. The DO generates some specific arguments like File ID, Owner ID, version no, timestamp, location of that block in the LA to the TPA. Upon receiving, the TPA creates the DLIT and the LA and also maintains them by completing all its pre-processing tasks.

2) *Verify Phase*: - This is the second phase where data verification is done by involving the DO, the CSP and the TPA. Challenges are send from the TPA to the CSP by using ChalGen and then the CSP responds by using ProofGen to prove the correctness of the data. In VerifyProof, the auditing results are submitted by the TPA to the DO.
a) ChalGen: - The TPA uses ChalGen to launch auditing challenges to the CSP. Then the TPA asks the CSP for the corresponding file and then verifies it by secret key. If it fails, the TPA informs to the DO, otherwise verification is done. Then, the TPA picks the random elements in the location array and gets those information from the DLIT. In short, the TPA sends auditing delegations to the CSP based on selected elements from the LA and the related blocks from the DLIT.

b) ProofGen: - The CSP uses ProofGen to generate auditing proofs for the particular blocks that can be sent by the TPA. ProofGen contains two parts namely, a tag proof indicates the correctness and other one called data proof indicates the data integrity. To be specific, these proofs are generated by the CSP. Upon the execution of ProofGen, the result of these proofs called complete proof will be sent back to the TPA as the response verification to the auditing challenges.

c) Verify Proof: - The TPA runs the VerifyProof to check the proofs returned from the CSP are valid or not. According to the information in the DLIT, the TPA computes DI for each data block in order to validate it. If it holds, the data outsourced to the cloud is complete, otherwise the data are incomplete.

Global and Sampling Verification:-Sampling Verification is already working on existing technologies, where the TPA randomly picks several data blocks to establish the challenge, in order to check the completeness and correctness of the data. But here, in this paper, the protocol is employed with a timestamp, which is generated by the DO and maintained by the TPA. The timestamp, which is automatically generated by the DO raises a problem i.e.., there is a possibility that DO will collude with the TPA by uploading a data block that will mismatch the timestamp. So, the uploaded data block is not going for data auditing. So, the DO and the TPA can carve up for compensation for data corruption from the CSP.

To solve these problems, global verification is addressed to stand against those collude attacks and also at the moment of data auditing any data block is updated or uploaded, all these data also be verified which comes under “global”. If the recently uploaded blocks are also passed for data auditing, then CSP will provide better services to its customers.In short, Sampling Verification solves the problems of the DO’s while the Global Verification serves the TPA.

Batch Auditing:-Batch Auditing allows the TPA to handle multiple auditing delegations from the various DO’s simultaneously. Batch Auditing is divided into two categories: one is to handle multiple auditing delegations from a single DO and the other one is to handle multiple delegations from the multiple owners.

DOI 10.17148/IJARCCE.2018.7313

Dynamic Auditing:-In this, some operations called block insertion, block deletion, block updating or modification are introduced. These operations are similar to the operations that can be done on a whole file.

Block Insertion: - Whenever a block b (i+1) is to be inserted after a block b (i), then the DO generates the version number, timestamp for the block b (i+1) to be inserted into the DLIT, as it is a new block. Moreover, DO sends the insertion instruction to the TPA. The insertion instruction to the TPA includes insert command, Owner ID, File ID, version no, block no, timestamp. Upon, receiving the request, the TPA updates the records in DLIT as well as LA. Meanwhile, the DO generates the corresponding authenticator for the block b (i+1) by sending the insertion instruction to the CSP then, the CSP inserts the new block. The insertion instruction to the CSP includes insert command, Owner ID, File ID, position, block b (i+1) and authenticator information. The insertion instruction is different for the TPA and the CSP.

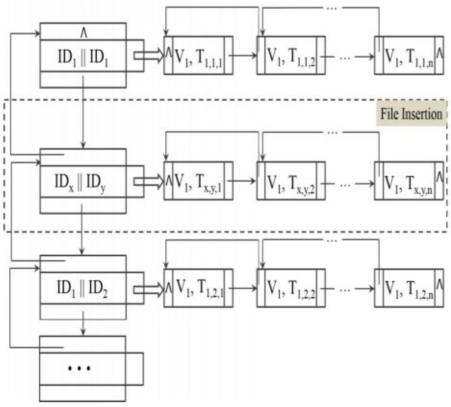


Fig 3.1: Block Insertion in DLIT

Insert element m_n

$$[a_{1,1}=m_1 \quad a_{1,2}=m_2 \quad a_{1,3}=m_n \quad a_{1,4}=m_3 \quad a_{1,5}=m_4 \ldots]$$

Fig 3.1.1: Block insertion in LA

Block Deletion: - This process is much more similar to the insertion process. But, it is simpler because, here the DO need not to generate new parameters like timestamp in the insertion process. The DO sends the delete instruction to the TPA. Upon receiving, the TPA searches for that particular block b (i) and then deletes its information in the DLIT as well as in LA. Moreover, the DO sends the delete instruction to the CSP, who deletes the data in that block. The delete instruction includes delete keyword, Owner ID, File ID, block b (i). The same instruction is send to the TPA and the CSP.

For suppose, the DO wants to delete the block m_2 from the file then, the changes made to the DLIT and the LA are shown in fig :3.2

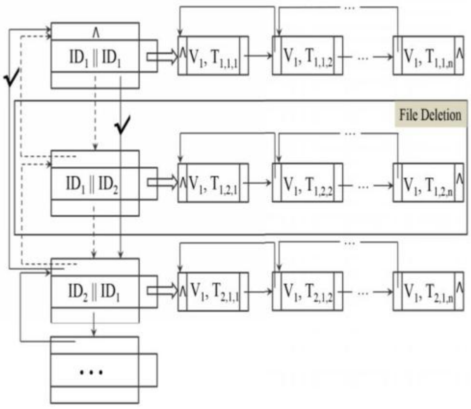


Fig3.2: Block Deletion in DLIT

Delete element m_2

$$[a_{1,1}=m_1 \quad a_{1,2}=m_n \quad a_{1,3}=m_3 \quad a_{1,4}=m_4 \ldots]$$

Fig 3.2.1: Block Deletion in LA

Block Modification: -To modify a block b (i), then the DO generates the corresponding version number and the timestamp for that block where the version number is incremented by one and the timestamp is independent of previous one. Then, the DO sends the modify instruction to the TPA which includes modify command, Owner ID, File ID, block no, version no and the timestamp. Upon, receiving the request, the TPA updates the information in the DLIT only but not in the LA, because the data in the block b (i) only changes, but not the sequential order of the blocks. Moreover, the DO generates a new authenticator for this modified block and sends the modify instruction to the CSP which includes modify command, Owner ID, File ID, block no, modified block and the new authenticator. On receiving the request, the CSP modifies the block as per the instructions. Here, the modify instruction is different for the DO and the CSP.

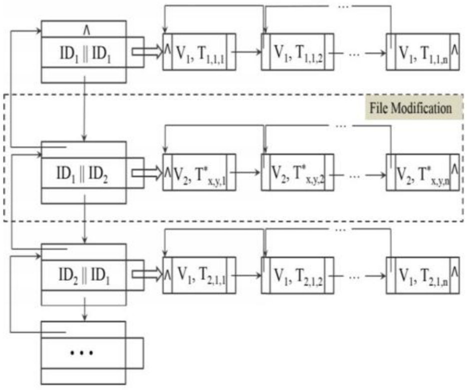


Fig3.3: Block Modification in DLIT

IV. EXPERIMENTALRESULTS

A. Data Owner

Data Owner owns the operations like Browse, Upload Blocks, Update, Verify and Delete.He has the flexibility to open and select a file in the local system and then upload it to the cloud by clicking Upload Blocks menu item. Data Owner has also a facility like deleting a file whenever he wishes by clicking Delete menu item. Meanwhile, DataOwner can verify the integrity and correctness of the data by using Verify option. Upon verifying, he has a facility to update the data, if he found any errors.Whatever, the file selected by the DataOwner to upload, it will open in the above display box.To achieve the functionalities like Upload Blocks, Update, Verify and Delete. Data Owner has to provide information about the particular filename, Block no's, IP address of the system he uses to perform these operations, which can be known by typing ipconfig command in the command prompt.For Example, Let us see the usage of Browse option in the DataOwner module, By clicking, on the Browse menu item, a small window will open which displays the files in the local system. Open button helps to choose files to upload, just uploading a file in gmail compose box.

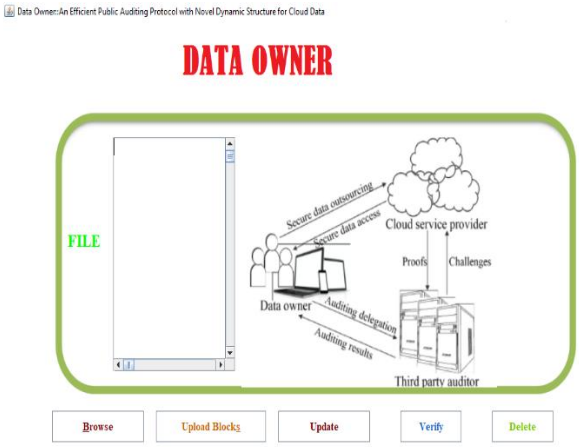
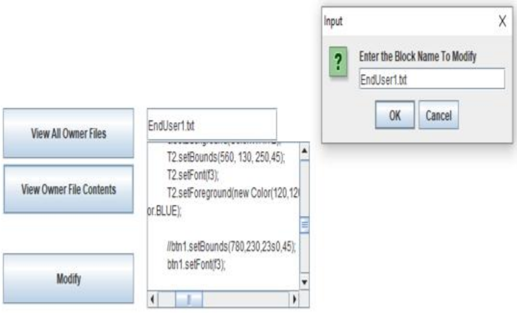


Fig 4:Data Owner

B.Cloud Server

Cloud Server is nothing but the Cloud Service provider. Cloud Service provider hasthe functionalities like View All Owner Files, View Owner File Contents and Modify.

DOI 10.17148/IJARCCE.2018.7313



EndUser1.txt

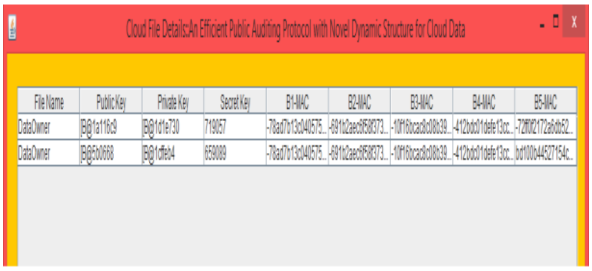
```
T2.setBounds(560, 130, 250,45);
T2.setFont(f3);
T2.setForeground(new Color(120,12
or.BLUE);

//btn1.setBounds(780,230,23s0,45);
btn1.setFont(f3);
```

Enter the Block Name To Modify

EndUser1.txt

Fig 5:Cloud Server



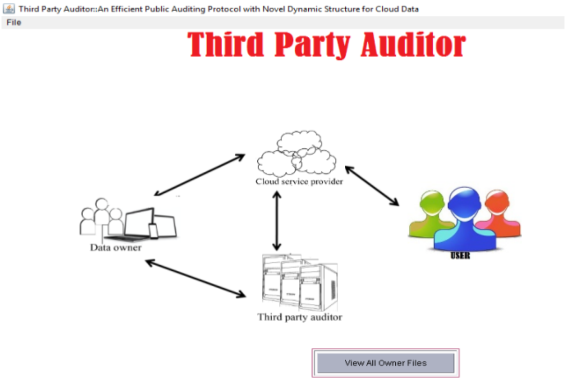
Cloud File Details:An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data

File Name	Public Key	Private Key	Secret Key	B1-MAC	B2-MAC	B3-MAC	B4-MAC	B5-MAC
DataOwner	[B@1a116c9	[B@1d1e730	719057	-78ad7b13c040575...	-691b2aecbf58f373...	-10f16bcac8c08b39...	-412bdd01defe13cc...	-72ff0f2172a6db52...
DataOwner	[B@5b0668	[B@1dffeb4	659089	-78ad7b13c040575...	-691b2aecbf58f373...	-10f16bcac8c08b39...	-412bdd01defe13cc...	bd100b44527154c...

Fig 6:View Owner Files

View All Owner Files displays a table that lists all the files uploa ded by the DO including their public, private and secret keys and also MAC addresses of the blocks.View Owner File Contents facilitates to diplay the contents of a particular block(s) in a file by giving the filename as input, as shown in the above fig 5.By default, the entire file is divided into 5(say) blocks. If we want to modify this number we can do it but, it is done during coding phase. We have to choose, what the number should be while implementation and we have to move on with it.Whatever, the modifications done to the data in the cloud by the DO will indirectly make an instruction to the CSP, just like TPA. Upon receiving the instructions, Cloud Server will move on as per the instructions.

C.Third Party Auditor

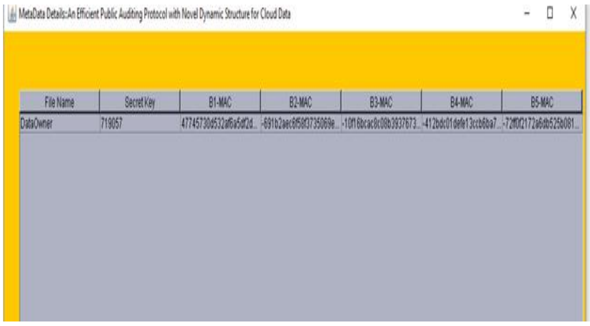


Third Party Auditor::An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data
File
Third Party Auditor
Cloud service provider
Data owner
USER
Third party auditor
View All Owner Files

Fig7: Third Party Auditor

Third Party Auditor is provided with a single functionality called View All Owner Files.TPA interacts while DO and CSP are communicating and also while CSP and user are communicating. While DO uploads the files and shares the files and also sends a request to TPA to check the integrity of the files, then TPA generates an auditing message to CSP and retrieve auditing proof from CSP. Then, TPA verifies correctness of the data auditing proof. Now, TPA sends an auditing report to the DO based on the result of verification.TPA has the talent to audit the already existing blocks in a file, including the blocks that are uploaded to that ongoing file which in turn facilitates Global Verification.The public auditing protocol in our paper allows an individual to verify the correctness of accessed data. It can verify the accessed data without looking what the data is.

DOI 10.17148/IJARCCE.2018.7313



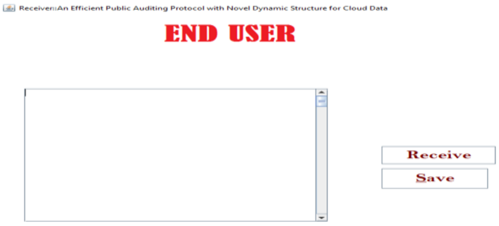
MetaData Details::An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data

File Name	Secret Key	B1-MAC	B2-MAC	B3-MAC	B4-MAC	B5-MAC
DataOwner	719057	47745730d532af6a5df2d...	-691b2aec6f58f3735069e...	-10f16bcac8c08b3937673...	-412bdc01defe13ccb6ba7...	-72ff0f2172a6db525b081...

Fig 8: View All Owner Files

TPA maintains a table which includes filenames, Secret Keys, MAC addresses of the five blocks to identify each block uniquely. Whenever TPA receives a request from the DO to verify the integrity and correctness of data, as it is provided with the knowledge of data verification and also TPA plays a crucial role in Block insertion, Block deletion and Block modification. TPA is the entity which handles the DLIT and the LA. Moreover, the modifications that had been done to DLIT and LA are only done by the TPA, upon receiving a request from the DO.The functionalities of the DO like Uploading, Deletion,and Verification of the Blocks will indirectly make an instruction to the TPA, as DLIT and LA are handled by the TPA, which handles the data in the blocks and on the other hand LA maintains the sequential order of these blocks. DLIT is more used than LA, because modifying the existing data in a particular block does not changes the order of the blocks.

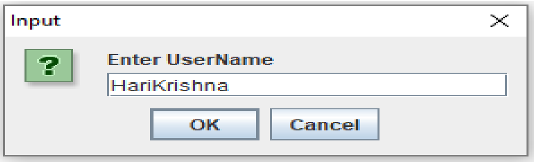
D.End User



Receiver::An Efficient Public Auditing Protocol with Novel Dynamic Structure for Cloud Data

Fig 9: End User

End User is provided with two functionalities namely Receive, Save.Receive functionality facilitates End Users to download a file by giving the file name, systems IP address, secret key of that file and more precisely Username has to be provided. Upon in taking all the inputs Cloud Server verifies them. If found any error, server is not going to provide any file to download. Otherwise, desired file is provided.Save functionality facilitates to save a particular file in the local system without asking any further details as Cloud Server verifies the user under receive functionality.

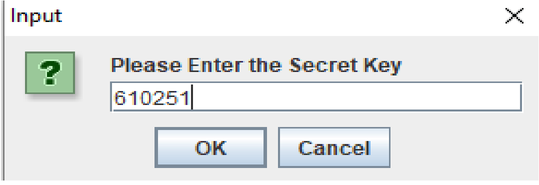


Input

Enter UserName

HariKrishna

Fig10.1:EndUsername

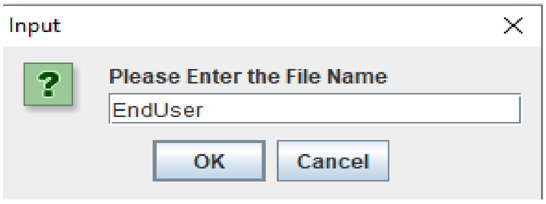


Input

Please Enter the Secret Key

610251

Fig 10.2 :Secret Key by DO

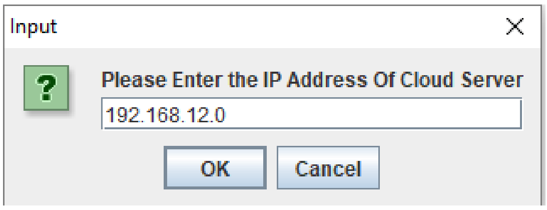


Input

Please Enter the File Name

EndUser

Fig 10.3 : File Name


Input

Please Enter the IP Address Of Cloud Server

192.168.12.0

Fig 10.4 : IP address of user's device
Fig 10: Sequence of inputs by End User

These are the sequence of inputs given by the End User to receive a file.End User will receive a file after validating all these inputs which is done behind the screen.

V. CONCLUSION

In this paper, we propose a public auditing protocol with novel dynamic structure composed of doubly linked info table and a location array. The appreciable relationship between the doubly linked info table and location array makes our protocol more efficient in terms of dynamic support and reduces block updating overheads. In addition to that, some additional features are provided like batch auditing, block-less verification and lazy update have been overcome by our protocol. Extensive numerical analysis, experimental comparison results are used to validate the performance of our protocol by making it more effective and convincing.

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing" Nat. Inst. Standards Technol, vol. 53, no. 6, p.50, 2011.
[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and L. Brandic, "Cloud Computing and emerging IT platforms: Vision, hype and reality for delivering computing as the 5Th utility," Future Generat.Comput. Syst., vol. 25, no.6, pp. 599-616, 2009.
[3] J. Shen, H. Tan, S. Moh, I. Chung, Q. Liu, and X. Sun, "Enhanced secure sensor association and key management in wireless body area networks," J.Commun. Netw., vol. 17, no. 5, pp. 453-462-2015.
[4] M. Green, "The threat in the cloud," IEEE Security Privacy, vol. 11, no. 1, pp. 86-89, Jan/Feb 2013.
[5] E. B. Dudin and Y.G. Smetanian, "A review of cloud computing," SciTech. inf. Process., vol. 38, no..4, pp.280-284, 2011.
[6] H. Tian et al., "Dynamic-hash table based public auditing for secure cloud storage," IEEE Trans. Serv. Compute., to be published.
[7] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. ACM Conf. Comput. Commun. Secur., 2007, pp. 598-609.
[8] J. Yu, K. Ren, and C. Wang, "Enabling cloud storage auditing with verifiable outsourcing of key updates," IEEE Trans. Inf. Forensics Security, vol. 11, no. 6, pp. 1167-1179, Jun. 2016.
[9] Y. Zhu, G.-J. Ahn, H. Hu, S. S. yau, H. G. An, and C.-J. Hu, "Dynamic audit services for outsourced storages in clouds," IEEE Trans. Serv. Comput., vol. 6, no. 2, pp. 227-238, Apr./Jun. 2013.
[10] H. Jin, H. Jiang, and K. Zhou, "Dynamic and public auditing with fair arbitration for cloud data," IEEE Trans. Cloud Comput., to be published.
[11] Jian Shen, Jan Shen & Xiaofeng Chen, Senior Memb, IEEE
[12] Xinyi Huang, Willy Sasilo, Senior Memb, IEEE.

DOI 10.17148/IJARCCE.2018.7313